Flow sensor (anemometer), 29008 Dofs, 1 to 3 parameters

Example with 1 parameter:

$$G(p) = c((sE - A_{v0} - p(A_{v1} - A_{v0}))^{-1}b)$$

with fluid velocity $p(= v) \in [0, 1]$ as single parameter and output as difference between the two sensors:

$$C(1, 173) = 1, \quad C(1, 133) = -1$$

permutation:

$$C = CP^T$$

for simulation: u is heater with step function (at id 35)

The following matrices are available:

*.B	load vector
*.E	damping matrix
*.C	permutation matrix (matrices built with ANSYS, for adapting nodes and dofs
*.A	stiffness matrices
_v0	diffusion
_v1	diffusion and convection

Use mmread for reading the matrices into matlab.

Example with 3 parameters: (infos about the matrices were extracted from IMTEK Mathematica code)

$$G(p_0, p_1, p_2) = c((s \underbrace{(M_s + p_0 M_f)}_{E(p_0)} - \underbrace{(A_{d,s} + p_1 A_{d,f} + p_2 A_c)}_{A(p_1,p_2)})^{-1} b)$$

with parameters:

$p_0 \in [0, 1]$	(c - specific heat)
$p_1 \in [0.1, 2]$	(κ - thermal conductivity)
$p_2 \in [1, 2]$	(cv - specific heat fluid velocity)
ρ	(mass density) seems to be fixed ($\rho = 1$?)

and matrices:

M_s	=	model_0_1_1.E
M_f	=	model_0_2_1.E - M_s
$A_{d,s}$	=	model_0_1_1.A
$A_{d,f}$	=	chop(model_1_1_1.A - $A_{d,s}$) + chop(model_1_2_2.A - model_1_2_1.A)
A_c	=	model_0_2_1.A - $A_{d,s}$
b	=	model_0_1_1.B

Infos from IMTEK: Take care of too small numbers (otherwise you will obtain illconditioned matrices) Thus, filter them out:

Mathematica: Chop[expr, delta] replaces numbers smaller in absolute magnitude than delta by 0. Chop uses a default tolerance of 10^{-10}
matlab (chop =):

$$A = A. * (abs(A) >= 10^{-12});$$